

Claims:

A listing of the claims has been included for reference. Please amend claims 1, 15, 23, and 31.

Listing of Claims:

1. (Currently Amended) A method, comprising:
collecting a loop trip count continuously during runtime of a region of code being executed that contains a loop;
categorizing the trip count to identify one or more code modification techniques applicable to the loop; and
dynamically applying, **during the same runtime**, the one or more applicable code modification techniques to alter the code that relates to the loop.

2. (Original) The method of claim 1, wherein collecting a loop trip count further comprises:
collecting a trip count for the loop each time the loop is entered; and
calculating an average trip count using a sequential plurality of collected trip counts over an interval of time.

3. (Original) The method of claim 2 further comprising executing the region of code for an introductory profiling phase time interval to establish an initial average trip count value.

4. (Original) The method of claim 3, wherein dynamically applying the one or more applicable code modification techniques further comprises applying one or more scalar transformation techniques to the loop upon receiving the initial average trip count value.

5. (Original) The method of claim 3, wherein categorizing the trip count further comprises:

determining a low trip count threshold value and a high trip count threshold value;
classifying the trip count as being in a first condition if the trip count is equal to or below the low trip count threshold value;

classifying the trip count as being in a second condition if the trip count is above the low trip count threshold value and below the high trip count threshold value; and

classifying the trip count as being in a third condition if the trip count is equal to or above the high trip count threshold value.

6. (Original) The method of claim 5 further comprising:
classifying the average trip count upon completion of each time interval subsequent to the introductory profiling phase to identify one or more loop transformation techniques applicable to the loop; and

dynamically applying the one or more applicable loop transformation techniques to alter the code that relates to the loop if the trip count classification changes.

7. (Original) The method of claim 6 further comprising instrumenting the code relating to the loop with one or more counters to monitor the loop trip count.

8. (Original) The method of claim 7 further comprising:
counting consecutive intervals of time that do not have a trip count classification change;

halting the trip count data collection if the number of consecutive intervals exceeds a threshold value; and

removing the one or more monitoring counters from the code relating to the loop.

9. (Original) The method of claim 3 further comprising:
determining if the loop has a regular control flow graph; and
applying one or more scalar transformation techniques to the code relating to the loop and one or more loop transformations to the code relating to the loop upon receiving the initial trip count value if the control flow graph is regular.

10. (Original) The method of claim 3 further comprising:
determining if the loop has substantial floating-point operations; and

applying one or more scalar transformation techniques to the code relating to the loop and one or more loop transformations to the code relating to the loop upon receiving the initial trip count value if the loop has substantial floating-point operations.

11. (Original) The method of claim 6, wherein applying loop transformations to the loop based on each trip count classification further comprises:

applying loop peeling and loop unrolling transformations to the loop if the trip count classification is in the first condition;

applying loop unrolling and software pipelining optimizations to the loop if the trip count classification is in the second condition; and

applying software pipelining and data prefetching optimizations to the loop if the trip count classification is in the third condition.

12. (Original) The method of claim 1, wherein collecting a loop trip count further comprises:

collecting a trip count for the loop each time the loop is entered; and

calculating an average trip count using a sequential plurality of collected trip counts over a determined number of iterations through the loop.

13. (Original) The method of claim 12, wherein the number of iterations through the loop is 50,000.

14. (Original) The method of claim 2, wherein the interval of time is equal to one second.

15. (Currently Amended) A method comprising,
repeatedly categorizing a loop trip count that is evaluated continuously during runtime;
determining after each categorization whether to apply one or more modification techniques to the loop if the categorization meets one or more criteria; and
dynamically applying, during the same runtime, the one or more applicable modification techniques to the loop based on the one or more criteria that are met.

16. (Original) The method of claim 15, wherein categorizing the loop trip count further comprises:
determining a low trip count threshold value and a high trip count threshold value;
classifying the trip count as being in a first condition if the trip count is equal to or below the low trip count threshold value;
classifying the trip count as being in a second condition if the trip count is above the low trip count threshold value and below the high trip count threshold value; and
classifying the trip count as being in a third condition if the trip count is equal to or above the high trip count threshold value.

17. (Original) The method of claim 16 further comprising:

classifying the average trip count upon completion of each time interval subsequent to the introductory profiling phase to identify one or more loop transformation techniques applicable to the loop; and

dynamically applying the one or more applicable loop transformation techniques to alter the code that relates to the loop if the trip count classification changes.

18. (Original) The method of claim 16, wherein the criteria further comprises whether the first condition is met.

19. (Original) The method of claim 16, wherein the criteria further comprises whether the second condition is met.

20. (Original) The method of claim 16, wherein the criteria further comprises whether the third condition is met.

21. (Original) The method of claim 15 further comprising:
counting consecutive intervals of time that do not have a trip count classification change; and

halting the trip count data collection if the number of consecutive intervals exceeds a threshold value.

22. (Original) The method of claim 21, wherein the criteria further comprises whether the threshold value has been exceeded.

23. (Currently Amended) A machine readable medium having embodied thereon instructions, which when executed by a machine, causes the machine to perform a method comprising:

collecting a loop trip count continuously during runtime of a region of code being executed that contains a loop;

categorizing the trip count to identify one or more code modification techniques applicable to the loop; and

dynamically applying, during the same runtime, the one or more applicable code modification techniques to alter the code that relates to the loop.

24. (Original) The machine readable medium of claim 23, wherein collecting a loop trip count further comprises:

collecting a trip count for the loop each time the loop is entered; and

calculating an average trip count using a sequential plurality of collected trip counts over an interval of time.

25. (Original) The machine readable medium of claim 24, wherein the method further comprises executing the region of code for an introductory profiling phase time interval to establish an initial average trip count value.

26. (Original) The machine readable medium of claim 25, wherein dynamically applying the one or more applicable code modification techniques further

comprises applying one or more scalar transformation techniques to the loop upon receiving the initial average trip count value.

27. (Original) The machine readable medium of claim 25, wherein categorizing the trip count further comprises:

determining a low trip count threshold value and a high trip count threshold value;
classifying the trip count as being in a first condition if the trip count is equal to or below the low trip count threshold value;
classifying the trip count as being in a second condition if the trip count is above the low trip count threshold value and below the high trip count threshold value; and
classifying the trip count as being in a third condition if the trip count is equal to or above the high trip count threshold value.

28. (Original) The machine readable medium of claim 27, wherein the method further comprises:

classifying the average trip count upon completion of each time interval subsequent to the introductory profiling phase to identify one or more loop transformation techniques applicable to the loop; and
dynamically applying the one or more applicable loop transformation techniques to alter the code that relates to the loop if the trip count classification changes.

29. (Original) The machine readable medium of claim 28, wherein the method further comprises instrumenting the code relating to the loop with one or more counters to monitor the loop trip count.

30. (Original) The machine readable medium of claim 29, wherein the method further comprises:

counting consecutive intervals of time that do not have a trip count classification change;

halting the trip count data collection if the number of consecutive intervals exceeds a threshold value; and

removing the one or more monitoring counters from the code relating to the loop.

31. (Currently Amended) A system, comprising:

a bus;

a processor coupled to the bus;

a network interface card coupled to the bus; and

memory coupled to the processor, the memory adapted for storing instructions, which upon execution by the processor collects a loop trip count continuously during runtime of a region of code being executed that contains a loop, categorizes the trip count to identify one or more code modification techniques applicable to the loop, and dynamically applies, during the same runtime, the one or more applicable code modification techniques to alter the code that relates to the loop.

32. (Original) The system of claim 31, wherein the system:
collects a trip count for the loop each time the loop is entered; and
calculates an average trip count using a sequential plurality of collected trip
counts over an interval of time.

33. (Original) The system of claim 32, wherein the system executes the
region of code for an introductory profiling phase time interval to establish an initial
average trip count value.

34. (Original) The system of claim 33, wherein the system applies one or
more scalar transformation techniques to the loop upon receiving the initial average trip
count value.

35. (Original) The system of claim 33, wherein the system:
determines a low trip count threshold value and a high trip count threshold value;
classifies the trip count as being in a first condition if the trip count is equal to or
below the low trip count threshold value;
classifies the trip count as being in a second condition if the trip count is above
the low trip count threshold value and below the high trip count threshold value; and
classifies the trip count as being in a third condition if the trip count is equal to or
above the high trip count threshold value.

36. (Original) The system of claim 35, wherein the system:

classifies the average trip count upon completion of each time interval subsequent to the introductory profiling phase to identify one or more loop transformation techniques applicable to the loop; and

dynamically applies the one or more applicable loop transformation techniques to alter the code that relates to the loop if the trip count classification changes.

37. (Original) The system of claim 36, wherein the system instruments the code relating to the loop with one or more counters to monitor the loop trip count.

38. (Original) The system of claim 37, wherein the system:
counts consecutive intervals of time that do not have a trip count classification change;
halts the trip count data collection if the number of consecutive intervals exceeds a threshold value; and
removes the one or more monitoring counters from the code relating to the loop.